

## PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2002-150281

(43)Date of publication of application : 24.05.2002

(51)Int.Cl.

G06T 3/40  
G06T 5/20  
H04N 1/393  
H04N 1/409

(21)Application number : 2000-349535

(71)Applicant : SONY CORP

(22)Date of filing : 16.11.2000

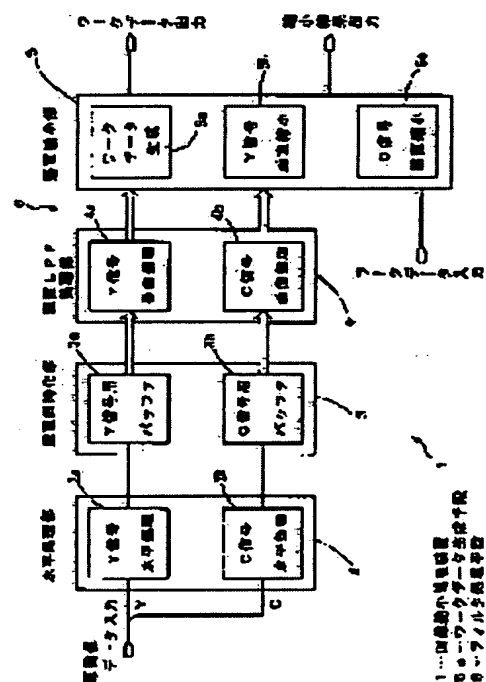
(72)Inventor : KITAMURA ATSUSHI  
TSUCHIDA HIROYASU

## (54) IMAGE COMPRESSION PROCESSING DEVICE AND METHOD

## (57)Abstract:

**PROBLEM TO BE SOLVED:** To provide an image compression device and method that are proper for a reduction in image memory capacity and a reduction in cost in image compression processing, and improve the speed of the processing.

**SOLUTION:** The image compression processing device 1 subjects original image data to filter processing and compression processing depending upon a given image reduction rate before outputting image data. The original image data are transferred in units of a plurality of segment blocks, and a filter processing means 6 is provided for horizontal or vertical filter processing on the data. The filter processing divides the horizontal or vertical filter into a plurality of segment filters so that an object block and adjacently located blocks can be each independently subjected to the filter processing. Out of pixel data after the filter processing, pixel data in a pixel position specified according to the image reduction rate are selectively outputted.



## LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's  
decision of rejection]

[Date of extinction of right]

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2002-150281

(P2002-150281A)

(43) 公開日 平成14年5月24日 (2002.5.24)

(51) Int.Cl. <sup>7</sup>	識別記号	F I	テ-マ-ド <sup>*</sup> (参考)
G 0 6 T	3/40	G 0 6 T 3/40	D 5 B 0 5 7
	5/20	5/20	A 5 C 0 7 6
H 0 4 N	1/393	H 0 4 N 1/393	5 C 0 7 7
	1/409	1/40	1 0 1 D

審査請求 未請求 請求項の数22 O L (全 21 頁)

(21) 出願番号 特願2000-349535(P2000-349535)

(22) 出願日 平成12年11月16日 (2000. 11. 16)

(71) 出願人 000002185

ソニー株式会社

東京都品川区北品川6丁目7番35号

(72) 発明者 喜多村 淳

東京都品川区北品川6丁目7番35号 ソニ  
ー株式会社内

(72) 発明者 土田 博康

東京都品川区北品川6丁目7番35号 ソニ  
ー株式会社内

(74) 代理人 100069051

弁理士 小松 祐治

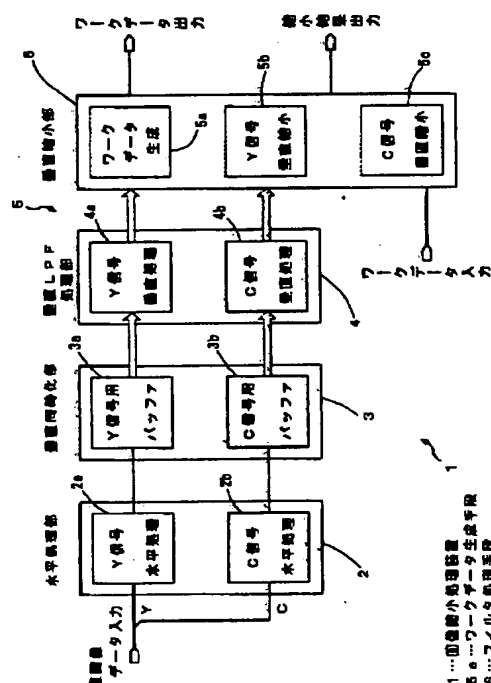
最終頁に続く

(54) 【発明の名称】 画像縮小処理装置及び方法

(57) 【要約】

【課題】 画像縮小処理時の画像メモリ容量の低減及び低コスト化に適した画像縮小装置及び方法を提供するとともに、処理速度の向上を図る。

【解決手段】 画像縮小処理装置1は、原画像データに対して所定の画像縮小率に応じたフィルタ処理及び縮小処理を施してから画像データを出力する。原画像データについては複数の分割されたブロックを単位としてデータが転送され、当該データに対して水平又は垂直方向のフィルタ処理を行うフィルタ処理手段6を設ける。フィルタ処理については、対象ブロック及びその近隣に位置するブロックについてそれぞれ各別にフィルタ処理を行うことができるように水平又は垂直方向のフィルタを複数の分割フィルタに区分けする。そして、フィルタ処理後の画素データについて画像縮小率に応じて規定される画素位置での画素データを選択的に出力する。



## 【特許請求の範囲】

【請求項 1】 原画像データに対して所定の画像縮小率に応じてフィルタ処理及び縮小処理が施された画像データを出力する画像縮小処理装置において、

上記原画像データについて複数の分割されたブロックを単位にしてデータが転送され、当該データに対して水平又は垂直方向のフィルタ処理を行うフィルタ処理手段を有し、

上記フィルタ処理手段が、対象ブロック及びその近隣に位置するブロックについてそれぞれフィルタ処理を行う複数の分割フィルタから成り、

フィルタ処理後の画素データについて画像縮小率に応じて画素選択が行われることを特徴とする画像縮小処理装置。

【請求項 2】 請求項 1 に記載した画像縮小処理装置において、

複数の分割フィルタについては、フィルタ処理に使用する多数の演算パターンが用意されており、縮小処理時には設定された画像縮小率に応じて演算に必要な演算パターンが選択されて各フィルタ毎に分割処理されるようにしたことを特徴とする画像縮小処理装置。

【請求項 3】 請求項 2 に記載した画像縮小処理装置において、

各分割フィルタによる分割処理時には、現ブロックよりも時間的に後に位置するブロック内の画素について演算に必要な演算結果を先行して算出しておくためのワークデータ生成手段を設けたことを特徴とする画像縮小処理装置。

【請求項 4】 請求項 3 に記載した画像縮小処理装置において、

ワークデータ生成手段によって現ブロックよりも先行して算出された演算結果を受けとって、これを現ブロック内の画素に係る演算結果と統合することにより最終的なフィルタ処理結果を出力する演算処理手段を設けたことを特徴とする画像縮小処理装置。

【請求項 5】 請求項 1 に記載した画像縮小処理装置において、

処理対象となるブロックが画像端の画素位置を含む場合に、当該画素位置の周囲にダミー画素を付設することにより画像周辺域を拡張する周辺拡張手段を設け、当該ダミー画素を含む隣接ブロック及び対象ブロックに対してそれぞれに対応する分割フィルタにより演算処理を行うことを特徴とする画像縮小処理装置。

【請求項 6】 請求項 2 に記載した画像縮小処理装置において、

処理対象となるブロックが画像端の画素位置を含む場合に、当該画素位置の周囲にダミー画素を付設することにより画像周辺域を拡張する周辺拡張手段を設け、当該ダミー画素を含む隣接ブロック及び対象ブロックに対してそれぞれに対応する分割フィルタにより演算処理を行う

ことを特徴とする画像縮小処理装置。

【請求項 7】 請求項 3 に記載した画像縮小処理装置において、

処理対象となるブロックが画像端の画素位置を含む場合に、当該画素位置の周囲にダミー画素を付設することにより画像周辺域を拡張する周辺拡張手段を設け、当該ダミー画素を含む隣接ブロック及び対象ブロックに対してそれぞれに対応する分割フィルタにより演算処理を行うことを特徴とする画像縮小処理装置。

10 【請求項 8】 請求項 4 に記載した画像縮小処理装置において、

処理対象となるブロックが画像端の画素位置を含む場合に、当該画素位置の周囲にダミー画素を付設することにより画像周辺域を拡張する周辺拡張手段を設け、当該ダミー画素を含む隣接ブロック及び対象ブロックに対してそれぞれに対応する分割フィルタにより演算処理を行うことを特徴とする画像縮小処理装置。

【請求項 9】 請求項 5 に記載した画像縮小処理装置において、

20 周辺拡張手段によって、画像端の画素位置の画素レベルと同じレベルのダミー画素を付設することを特徴とする画像縮小処理装置。

【請求項 10】 請求項 6 に記載した画像縮小処理装置において、

周辺拡張手段によって、画像端の画素位置の画素レベルと同じレベルのダミー画素を付設することを特徴とする画像縮小処理装置。

【請求項 11】 請求項 7 に記載した画像縮小処理装置において、

30 周辺拡張手段によって、画像端の画素位置の画素レベルと同じレベルのダミー画素を付設することを特徴とする画像縮小処理装置。

【請求項 12】 請求項 8 に記載した画像縮小処理装置において、

周辺拡張手段によって、画像端の画素位置の画素レベルと同じレベルのダミー画素を付設することを特徴とする画像縮小処理装置。

【請求項 13】 請求項 1 に記載した画像縮小処理装置において、

40 1 ブロックに係る水平幅を画素単位で「a」個とし、1 ブロックに係る垂直幅を画素単位で「b」個とすると、 $a \times (b - 1)$  個分の画素データを保持する同時化手段を、垂直方向におけるフィルタ処理手段の前に配置したことを特徴とする画像縮小処理装置。

【請求項 14】 原画像データに対して所定の画像縮小率に応じてフィルタ処理及び縮小処理が施された画像データを取得するための画像縮小処理方法において、

上記原画像データについて複数の分割されたブロックを単位にしてデータを転送し、

50 当該データに対して水平又は垂直方向のフィルタ処理を

行う際には、対象ブロック及びその近隣に位置するブロックについてそれぞれのフィルタ処理を行う分割フィルタを用いて演算を行った後、フィルタ処理後の画素データについて画像縮小率に応じて規定される画素位置でのデータを縮小データとして出力するようにしたことを特徴とする画像縮小処理方法。

【請求項 15】 請求項 14 に記載した画像縮小処理方法において、

複数の分割フィルタについて、フィルタ処理に使用する多数の演算パターンを予め用意しておき、その後の縮小処理時には、設定された画像縮小率に応じて演算に必要な演算パターンを選択して各フィルタ毎に分割処理することを特徴とする画像縮小処理方法。

【請求項 16】 請求項 15 に記載した画像縮小処理方法において、

各分割フィルタによる分割処理時には、現ブロックよりも時間的に後に位置する後続ブロック内の画素について演算に必要な演算結果を先行して算出しておき、その後に、上記後続ブロック内の画素について演算を行う際には、当該ブロックよりも先行して算出されている上記演算結果を受けとって、これを対象ブロック内の画素に係る演算結果と統合することにより最終的なフィルタ処理結果を出力するようにしたことを特徴とする画像縮小処理方法。

【請求項 17】 請求項 14 に記載した画像縮小処理方法において、

処理対象となるブロックが画像端の画素位置を含む場合に、当該画素位置の周囲にダミー画素を付設して、当該ダミー画素を含む隣接ブロック及び対象ブロックに対してそれぞれに対応する分割フィルタにより演算処理を行うことを特徴とする画像縮小処理方法。

【請求項 18】 請求項 15 に記載した画像縮小処理方法において、

処理対象となるブロックが画像端の画素位置を含む場合に、当該画素位置の周囲にダミー画素を付設して、当該ダミー画素を含む隣接ブロック及び対象ブロックに対してそれぞれに対応する分割フィルタにより演算処理を行うことを特徴とする画像縮小処理方法。

【請求項 19】 請求項 16 に記載した画像縮小処理方法において、

処理対象となるブロックが画像端の画素位置を含む場合に、当該画素位置の周囲にダミー画素を付設して、当該ダミー画素を含む隣接ブロック及び対象ブロックに対してそれぞれに対応する分割フィルタにより演算処理を行うことを特徴とする画像縮小処理方法。

【請求項 20】 請求項 17 に記載した画像縮小処理方法において、

画像端の画素位置の画素レベルと同じレベルのダミー画素を付設することを特徴とする画像縮小処理方法。

【請求項 21】 請求項 18 に記載した画像縮小処理方

法において、

画像端の画素位置の画素レベルと同じレベルのダミー画素を付設することを特徴とする画像縮小処理方法。

【請求項 22】 請求項 19 に記載した画像縮小処理方法において、

画像端の画素位置の画素レベルと同じレベルのダミー画素を付設することを特徴とする画像縮小処理方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、画像縮小処理のために必要なメモリ容量を減らし、かつ処理速度を向上させるための技術に関するものであり、ブロック状にまとまって転送されてくる原画像データを縮小処理する際に使用するフィルタを複数に分割して各フィルタによる演算結果（ワークデータ）を入出力しながら、縮小後に必要な画素位置データについての演算結果を出力するように構成したものである。

【0002】

【従来の技術】画像信号処理において、画素の転送方法として所謂ラスタースキャン方式（つまり、画面最上ラインの左端画素から右端へ向かって1画素ずつ走査し、1ラインの終了後には当該ラインの下ラインについてその左端から右端へ向かって1画素ずつ走査するという具合に、当該処理を画面の最下ラインに到達するまで繰り返すことにより、画素データを順次に取り出して出力する転送形態であり、例えば、CCD型イメージセンサー等の固体撮像素子における画素出力に用いられる。）のシステムでは、 $n$ 次のフィルタ処理を画面の水平方向及び垂直方向で行う場合に、水平画素数と「垂直ライン数-1」との積に相当する画素分の情報を、メモリ等の記憶手段によって一時的に保持しておき、その記憶データに対してフィルタ処理を施した出力を得る構成となっている。

【0003】特に、画像情報の保持手段として、多くの場合、1画面分の全てのデータを画像メモリに一旦格納しておいてから、その読み出し方向を、水平方向、垂直方向でそれぞれ入れ替えることで水平垂直処理を行っている。

【0004】図16は従来の処理例についての概念的な説明図であり、メモリa内に格納された画像データを水平方向（図に矢印「H」で示す。）に沿って読み出して処理する水平処理部bと、画像データを垂直方向（図に矢印「V」で示す。）に沿って読み出して処理する垂直処理部cとが設けられている。

【0005】

【発明が解決しようとする課題】しかしながら、従来の装置にあつては、取り扱う画像の画素サイズが大きい場合に、これに伴ってメモリ容量を増加しなければならないので、回路規模が大きくなり、コスト高の原因となったり、装置の小型化を妨げる等の問題がある。

【0006】そこで、画像情報をブロック状に分割してから転送を行う処理方式（以下、「ブロック処理」という。）が考えられ、このような画像処理系では、画像全体のデータから必要な部分を読み出す手法を採っており、例えば、画像データを複数の単位画像データ群（ブロック）に分割して、各ブロックのうちから処理に必要なブロックのデータを読み出すことにより、メモリ容量の小規模化等を含む各種の利点が得られるが、画像縮小処理だけではそのような利点を十分に生かすことができず、処理速度に問題が残されている。

【0007】本発明は、画像縮小処理時の画像メモリ容量の低減及び低コスト化に適した画像縮小処理装置及び方法を提供するとともに、処理速度の向上を図ることを課題とする。

【0008】

【課題を解決するための手段】本発明は上記した課題を解決するために、下記に示す事項を採用したものである。

【0009】・原画像データについては複数の分割されたブロックを単位にしてデータが転送され、当該データに対してフィルタ処理手段により水平又は垂直方向のフィルタ処理を行うこと。

【0010】・フィルタ処理については、対象ブロック及びその近隣に位置するブロックについてそれぞれ各別にフィルタ処理を行うことができるように水平又は垂直方向のフィルタを複数の分割フィルタに分けたこと。

【0011】・フィルタ処理後の画素データについて画像縮小率に応じて規定される画素位置での画素データが選択出力とされること。

【0012】従って、本発明によれば、ブロック処理の採用により容量の大きなメモリを必要としないことは勿論、現ブロック及びその近隣ブロックに対するフィルタの分割処理によって高速な画像縮小処理を実現することができる。

【0013】

【発明の実施の形態】本発明に係る画像縮小処理装置は、原画像データに対して指定された所定の画像縮小率に応じてフィルタ処理及び縮小処理が施された画像データを出力するための装置であるが、その構成について説明する前に、本発明における原画像データの入力方法及び処理形態について図1を用いて説明する。

【0014】本発明では、ブロック処理を基本とし、図1に示すように、水平方向「H」及び垂直方向「V」に沿って「 $a \times b$ 」画素からなる小ブロックに分割して、このブロックを単位としてデータ転送を行う。尚、「 $a$ 」は小ブロックの水平方向における画素数（水平幅）を示し、また、「 $b$ 」は小ブロックの垂直方向における画素数（垂直幅）を示している。よって、水平方向における画素数を「 $X$ 」とすると当該方向に沿って「 $X/a$ 」個の小ブロックが配置され、垂直方向における画

素数を「 $Y$ 」とすると当該方向に沿って「 $Y/b$ 」個の小ブロックが配置される。

【0015】各ブロック内の画像データについては、例えば、画面上で最も左上に位置する小ブロック（「Block ID 0」を参照。）内に示した複数の矢印から分かるように、当該ブロック内の最上ラインについて最左端画素から図の右方へと1画素ずつ転送し、当該ブロック内の最右端画素まで転送すると、現ラインよりも1ライン下の最左端画素に移って上記と同じ処理を繰り返していく。そして、1ブロック全部のデータ転送が終了したらその右隣のブロック（小ブロック「Block ID 1」を参照。）へと移動して、前ブロックと同じ処理を行う。このような処理が繰り返されて、対象となるブロックが画面の最右端にまで到達し、当該ブロックについて処理を終了した後は、現ブロックよりも1つ下のブロックの並びにおける最左端ブロック（小ブロック「Block ID  $X/a$ 」を参照）へと移動して再び同じ手順を繰り返す。

【0016】以下では、水平方向に沿う横並び1行を「ブロックライン」と呼ぶことにし、各ブロックラインに対して識別子「Block Line」と数字とからなる識別符号を付して区別する。例えば、画面の最上位置のブロックラインを「Block Line 0」とし、その下（V方向における下側）のブロックラインを「Block Line 1」という具合に「0」を起点として、「 $Y/b-1$ 」を終点とする番号のブロックラインが垂直方向に沿って配列される。

【0017】また、個々のブロックについて区別するために、それらに識別子「Block」と数字とからなる識別符号を付すことにする。例えば、ブロックライン「Block Line 0」（0乃至「 $b-1$ 」ライン）において、最左端に位置する小ブロックに対して「Block ID 0」を、その右隣の小ブロックに対して「Block ID 1」という具合に番号を1ずつ増やして付与していくと、「Block Line 0」の最右端に位置する小ブロックは「Block ID  $X/a-1$ 」となる。同様にして、「Block Line J」（Jは自然数変数であり、このブロックラインは「 $J \cdot b$ 」乃至「 $(J+1) \cdot b-1$ 」ラインを含む。）については、最左端から「Block ID J・ $(X/a)$ 」乃至「Block ID  $(J+1) \cdot (X/a)-1$ 」の小ブロックから成ることが分かる。そして、数字が増大する方向に沿って1ブロックずつデータ転送が行われる。

【0018】以上のような転送方法を採用することの利点は、例えば、JPG（Joint Photographic Experts Group）方式での圧縮・伸長のように、 $8 \times 8$ 画素単位で画像処理を行う必要がある回路の直後において、メモリへの書き出しや、読み出しの処理を必要としないため、メモリ容量が最小限で済み、また処理速度を高速化

できることにある。即ち、処理部の外部メモリ（DRAM等）に処理結果を書き出して保持したり、改めて縮小ブロックに都合の良いように処理結果を読み出す必要がなくなり、本発明に係る縮小処理回路（後で詳述する。）を回路前後に配置するだけ済むようになる。

【0019】次に、画像縮小処理について説明する。

【0020】今、仮に小ブロックが「 $32 \times 8$ 」画素を単位として構成されており、全画像、つまり、 $X$ 個 $\times Y$ 個の画素からなる原画像を縦横同率（例えば、2分の1）で縮小する場合を考える。

【0021】処理手順は、概ね以下の通りである。

【0022】（1）ローパスフィルタ（LPF）を通して画像信号の広域成分を減衰させる

（2）設定された縮小率に応じて画素を抜き出す。

【0023】尚、手順（1）を必要とする理由については、当該手順を経ることなく単に手順（2）により適当な画素を間引いたのでは、縮小後の画像に折り返しノイズが発生し、画質劣化を招く虞があること等に依る。

【0024】画像縮小処理には水平方向の縮小処理と、垂直方向の縮小処理とが含まれるので、以下、この順序に従って説明する（尚、水平処理について、後述するフィルタ分割の方法には触れることなく説明するが、これは理解のし易さを考慮した説明の便宜によるものである。）。

【0025】先ず、水平方向における上記手順（1）において、15タップ（回数に相当する。）のローパスフィルタを用いる場合を想定する。

【0026】1つの小ブロックについて水平方向の画素数を32個としているので、ある小ブロック内のある画素について15タップのフィルタ処理をかけるには、当該画素を含む近隣画素のデータとして15個のデータが必要となるが、これはほとんどの場合に調達可能である（画面の周辺域に位置する小ブロック内では、必要数の画素データが集められない場合が生じるが、これについては後述の画端処理の説明において詳述する。）。つまり、ある画素の近隣画素を含む画素群のデータに基づいて各データに対する加減算、係数乗算等によるフィルタ処理（あるいはフィルタリング処理）を行うことができる。

【0027】尚、フィルタ処理については、輝度（Y）信号と色（C）信号に分けてローパスフィルタを設けるとともに、各フィルタの回数についても格別に設定することが好ましい。

【0028】次に、上記手順（2）について説明する。

【0029】図2及び図3は縮小処理における選択画素についての説明図であり、図2が縮小率「 $1/2$ 」の場合を示し、図3が縮小率「 $1/2.5$ 」の場合を示している。

【0030】図2において、水平方向の8画素には左端から右方にかけて1乃至8の番号をそれぞれ付してい

る。本例では縮小率が「 $1/2$ 」であり、奇数番号「1」、「3」、「5」、「7」が抽出されて画素選択がなされる。尚、1ブロックの水平幅として32画素を想定しているので、実際には1乃至32までの間の奇数番号を付した画素が選択される。

【0031】また、図3には水平方向の20画素を示しており、各画素には左端から右方にかけて1乃至20の番号を付している。本例では縮小率が「 $1/2.5$ 」であり、20画素のうち、番号「1」、「4」、「6」、「9」、「11」、「14」、「16」、「19」の8個が抽出されて画素選択がなされる（「 $20/2.5=8$ 」）。つまり、番号1を起点として、1つ飛び越した「3」の後ろに位置する画素「4」が選択され、さらに1つ飛び越した画素「6」が選択される。そして、それから1つ飛び越した「8」の後ろに位置する画素「9」が選択されるという具合にして上記の選択画素列が得られる。尚、この処理は、抽出パターンを「（1，0，0，1，0）」と記す時、論理値「0」に対応する画素を無視し、論理値「1」に対応する画素を抜き出す操作（5個を単位にその中から2つ選び出すこと。）と等価であり、例えば、画素列「6」乃至「10」に対して上記の抽出パターンを適用すると、画素「6」、「9」が選択されることが分かる。勿論、抽出パターンとして別のものを使用すれば（例えば、「（1，0，1，0，0）」等。）、異なる結果が得られる。

【0032】このように、縮小率「 $1/2.5$ 」の場合に、1ブロックの水平幅として32画素を想定したときの間隔「2.5」の数値「1-3.5-6-8.5-11-13.5-16-18.5-21-23.5-26-28.5-31」において整数でないもの（下線部）については切り上げ又は四捨五入により整数化すれば、「1-4-6-9-11-14-16-19-21-24-26-29-31」となって上記と同じ結果が得られることが分かる。但し、この場合には、隣接ブロックにおいて「 $31+2.5=33.5=32+1.5$ 」となり、33番目の画素を起点として1.5から始まるパターン等が必要となることが容易に理解される（つまり、上記数値に対応する1通りのパターンを各ブロックについてただ繰り返すだけでは済まないという事情がある。）。

【0033】このようなずれを考慮して画素位置の数値を列記すると、上記の他に下記の4パターンが挙げられる。

【0034】・1.5（2）-4-6.5（7）-9-11.5（12）-14-16.5（17）-19-21.5（22）-24-26.5（27）-29-31.5（32）  
・2-4.5（5）-7-9.5（10）-12-14.5（15）-17-19.5（20）-22-24.5（25）-27-29.5（30）-32

・ 2. 5 (3) - 5 - 7. 5 (8) - 10 - 12. 5  
 (13) - 15 - 17. 5 (18) - 20 - 22. 5  
 (23) - 25 - 27. 5 (28) - 30  
 ・ 0. 5 (1) - 3 - 5. 5 (6) - 8 - 10. 5 (11) - 13 - 15. 5 (16) - 18 - 20. 5 (21) - 23 - 25. 5 (26) - 28 - 30. 5 (31)

尚、括弧内に示す数字は、整数化後の数字を示しており、これについては後述するようにサブフィルタを通したものを採用する。

【0035】縮小率「1/3」の場合には、画素位置の数列について下記の3パターンが挙げられる。

【0036】・ 1 - 4 - 7 - 10 - 13 - 16 - 19 - 22 - 25 - 28 - 31  
 ・ 2 - 5 - 8 - 11 - 14 - 17 - 20 - 23 - 26 - 29 - 32  
 ・ 3 - 6 - 9 - 12 - 15 - 18 - 21 - 24 - 27 - 30

尚、同一ブロック内では同じパターンを使用する。そして、あるブロックについて、例えば、上記の場合に一番上のパターンを使用したときには、1つ右側のブロックでは2番目のパターンを使用することになる（「31 + 3 = 34 = 32 + 2」より「2」が起点となるから）。

【0037】上記した画素選択の基本アルゴリズムは、次の通りである。

【0038】・ 1画素1ライン目を基準とする。例えば、縮小率1/2や1/4等では、最初の画素が出力され、また、その他の場合（縮小率1/3等）でも画素選択の繰り返しのパターンにおいて最初のパターンの開始点に位置する画素が必ず出力される（つまり、上記した抽出パターンや画素位置の数列を適用する場合において、それらの開始位置における画素が選択されるということである。）。

【0039】・ 出力（選択）すべき画素に付与された番号（画素番号）が整数値の場合には、これをそのまま（加工せずに）抜き出す。例えば、縮小率「1/2」の場合では、画素番号として奇数番号の画素が選択されるので、それらの番号は全て整数である。

【0040】・ 出力すべき画素に付与された画素番号が整数値でない場合には、当該番号に隣接する画素番号のうちのいずれか一方を選択する。例えば、縮小率を「1/2. 5」とした上記の場合では、画素番号「1」、「3. 5」、「6」、「8. 5」、…を本来とり出すべきであるが、「3. 5」や「8. 5」等のように、小数「0. 5」をもつ数字に対応する画素（0. 5ラインずれた画素位置）は存在しないので、小数「0. 5」を切り上げ若しくは四捨五入した数字に対応する番号の画素を有効としてそれらを選択する（2画素の中心位置に重心をもつ画素情報をフィルタ生成するのに、2画素分の

情報がくるのを待つ必要性から。）。

【0041】次に垂直方向の縮小処理について説明するが、小ブロックの垂直方向における画素数を8個としているため、上記手順（2）において、タップ数の多い、長いフィルタをかける必要がある場合にそのままでは不都合が生じることになる。つまり、垂直方向において1つブロックには8画素分しか存在しないので、15タップを埋めるだけの画素データを1ブロックだけでは調達できない。

【0042】そこで、フィルタを複数個に分割することにより、垂直方向において最大で3つのブロックに跨る演算パターンを用意する。

【0043】図4はそのような分割例を示したものであり、縮小率「1/2」を想定している。

【0044】図示する各パターン1乃至4において、最も上方に示すものが垂直方向において2ブロック先（つまり、2ブロックライン先）のj（=1、3、5、7）ライン目の処理に必要な画素位置を示しており、例えば、パターン1（j=1）ではアルファベット「a」乃至「f」を付した6画素（最上の2ラインに位置する画素を除いた画素）について演算が行われる。尚、同じアルファベットを付したのもの同士については同じフィルタ係数（重み付け係数）が割り当てられる。

【0045】また、同図の中間に示す8画素については、垂直方向において1ブロック先（つまり、1ブロックライン先）のjライン目に処理に必要な画素位置を示している。例えば、パターン1では、上から順に「g」、「h」、「g」、「f」、「e」、「d」、「c」、「b」が付されており、これらは全て1ブロック先の1ライン目に処理に必要な画素位置を示す。

【0046】そして、図で最も下方に示すものが、現ブロックにおいてjライン目の処理に必要な画素位置を示しており、当該ブロック内において「a」で示すもの（丸印で囲んで示す。）が処理対象となる画素位置を示している。例えば、パターン1では、最上ライン（1ライン目）に位置する画素が対象とされる。

【0047】これらのパターンは垂直方向においてフィルタ処理に必要な画素配列だけを示したものであり、例えば、図1において小ブロック「BlockID」・（X/a）」内に位置する画素を対象とする場合には、当該小ブロックの上に位置する2つの小ブロック「BlockID（J-1）・（X/a）」、「BlockID（J-2）・（X/a）」内において対象画素と同列に位置する画素データが必要となる。

【0048】図4のパターン2は、パターン1の各要素を2単位（2画素分）だけ図の下方にずらす（垂直方向のシフト）操作によって得られる。同様に、パターン3はパターン2の各要素を2単位だけ下方にずらすことで得られ、パターン4はパターン3の各要素を2単位だけ下方にずらすことで得られる（この場合にはパター



ンが2ブロックに跨っている。)。尚、2画素分のシフト操作については、縮小率として「1/2」を想定しており、垂直方向に沿って1つおきに画素選択が行われることに依る。また、画面の周辺域に位置する小ブロック内の画素では、必要数の画素データが集められない場合が生じるが、これについては後述の画端処理の説明において詳述する。

【0049】上記のように、小ブロックの垂直方向における幅を超えるタップ数のフィルタ処理を行う場合、例えば、幅8画素分に対して15タップフィルタでは、最大で3ブロックに亘る処理が必要となる。しかしながら、その処理のために、現ブロックの上側2ブロックライン分に相当する原画像データを蓄積するメモリを使用したのでは、メモリ容量が増大してしまう。そこで、これを避けるためには、フィルタ処理自身を複数に分割して、各フィルタ（分割フィルタ）による処理結果を現ブロックの処理のためのワークデータとして出力し、現ブロックでそれらのワークデータを受け取ってフィルタリングの演算結果を取得する方法が望ましい。

【0050】つまり、図4の例では、フィルタを最大で3つに分割し（パターン4については2分割）、2つ上に位置するブロック内で必要な画素データから得られる演算結果や、1つ上に位置するブロック内で必要な画素データから得られる演算結果を求めておき（演算を先行して行う。）、現ブロックでは、これらの演算結果を受け取って最終的なフィルタリング演算を行ってその結果を出力するように回路を組めば、現ブロックラインに隣接してその上方に位置する2ブロックライン内の全画像データをメモリに格納しておく必要は全くなくなる。例えば、図4のパターン1について、具体的に説明すると、現ブロックよりも2ブロック上に位置するブロック内で必要な画素を、現ブロックに近い方から順にリスト表記で示すと「(f, e, d, c, b, a, 0, 0)」(「0」は不要画素あるいは演算に関与しない画素を意味する。)となり、それらの画素位置に対応するデータから演算結果（これを「WK1」と記す。）を求めることができる。同様にして、現ブロックよりも1ブロック上に位置するブロック内で必要な画素を、現ブロックに近い方から順にリスト表記で示すと「(b, c, d, e, f, g, h, g)」となり、それらの画素位置に対応するデータから演算結果（これを「PWK2」と記す。）を求めることができるとともに、この時点では当該演算結果PWK2と上記WK1とを加算して得られる中間的な演算結果（これを「WK2」と記す。）を得ることができる。よって、現ブロックの1ライン目に位置する画素「a」に来たときには、リスト表記「(0, 0, 0, 0, 0, 0, 0, a)」に対応するデータから演算結果（これを「WK3」と記す。）を求めるとともに、これに対して、それ以前の演算により先行して計算しておいた上記WK2を加算することにより、現ブロッ

ク内の画素「a」についての最終演算結果（これを「SCL」と記す。）を出力すれば良い。他のパターンについても同様の手続きが行われ、例えば、パターン3では、(b, a, 0, 0, 0, 0, 0, 0)に対する演算結果としてWK1が、(f, g, h, g, f, e, d, c)に対する演算結果としてPWK2が算出され、「WK2 = PWK2 + WK1」からWK2が求まる。そして、(0, 0, 0, a, b, c, d, e)に対する演算結果WK3が算出されてから、「SCL = WK3 + WK2」によりSCLが求まる。

【0051】尚、本発明に関する限り、フィルタ処理内容の如何は全く問わないので、例えば、上記した最大3ブロックに跨る演算処理として、単純平均計算を用いる場合には、上記リスト表記中で「0」でない画素位置でのデータを加算した後で平均化する（あるいは所定の係数を掛けた後に加算する）処理を行うものと考えれば、上記した説明内容について理解し易い。

【0052】図5は15タップフィルタについて現ブロック内の8画素が採り得る位置の組み合わせの数をすべて網羅した演算パターンを示したものであり、同図の上段に示すものが上記「WK1」の演算に必要なパターン（群）を、中段に示すものが上記「PWK2」の演算に必要なパターン（群）を、下段に示すものが上記「WK3」の演算に必要なパターン（群）をそれぞれ示している。尚、図中に示す「CPz」（z=0, 1~23）は、垂直8画素を単位とする24通り（CP0とCP15、CP7とCP23を重複パターンとして取り除いて数えた場合には22通り）のパターンに対してそれぞれ付与した識別符号であり、その中でCP0、CP7は0.5ライン目の出力処理において必要になる。また、「Ft」（t=0, 1~14）は15タップフィルタについて各タップを埋めるべき画素位置を示しており（各タップの係数に対応する。）、「F14」が出力対象となる画素位置を示している。本図における各演算パターンと、図4のパターンとの関係については、図5の上下方向が垂直方向に相当することから分かるように、CP1、CP8、CP16がパターン1に相当し、CP3、CP10、CP18がパターン2に相当する（説明は省略するがその他についても同様である。）。

【0053】以上のように、図5は、8画素を単位とする入力情報に対して15タップの係数からその8個をどの画素位置に割り当てるかに応じたパターンとして得られるものであり、よって、さほど複雑な手続きを経ることなく作成することができ、これらの演算パターンに相当する回路を事前に用意しておくことができる。尚、前記した水平方向処理に係るフィルタ処理についても、1ブロックの水平方向の幅を、8画素程度に狭くした場合には、上記と同様にフィルタ分割の方法を用いることができる（例えば、1ブロックの水平幅に比して水平方向LPFのタップ数が多く、フィルタの分割数が多い場合

や、画面の転送方法について、図1の水平方向と垂直方向を入れ替えた場合等において、上記したフィルタ分割による演算方法は効果的である。)

【0054】垂直方向における上記手順(2)については、水平方向の縮小処理の場合と基本的には同様である。つまり、図2、図3について行った説明において、「水平方向」を「垂直方向」に適宜読み替えれば良いが、垂直方向の処理では、上記したように処理(垂直LPF処理)に必要なワークデータ(WK1、WK2等)を作成しながらフィルタ演算処理を行っていくことを考\*10

1ライン目	3ライン目
WK1: CP1	CP3
PWK2: CP8	CP10
WK3: CP16	CP18

つまり、ライン番号が大きくなる方向が時間の経過方向に対応するので、上記において、1ライン目の縦列に示す演算パターンが図4のパターン1、3ライン目の縦列に示す演算パターンがパターン2、5ライン目の縦列に示す演算パターンがパターン3、7ライン目の縦列に示す演算パターンがパターン4にそれぞれ対応することが分かる。尚、「Dummy」は、該当する演算パターンがないこと(あるいは上記したリスト表記で全要素が「0」であること。)を意味する。

【0058】また、縮小率「1/4」の場合に使用する演算パターンは下記の通りである。 ※

・1、4、7ラインを出力する場合

演算パターン

WK1: CP3	CP6
PWK2: CP9	CP12
WK3: CP16	CP19

・2、5、8ラインを出力する場合

演算パターン

WK1: CP1	CP4
PWK2: CP10	CP13
WK3: CP17	CP20

・3、6ラインを出力する場合

演算パターン

WK1: CP2	CP5
PWK2: CP8	CP11
WK3: CP18	CP21

尚、この場合には、垂直方向の1ブロック当たり1ラインずつのずれがあることに注意を要する。即ち、縮小率「1/2」のときのように、WK1、PWK2、WK3についてすべて同じライン(1、3、5、7)を生成するための演算パターンを選択すれば良いのではなく、あるブロックの1、4、7ラインについてフィルタ出力を得ようとする場合には、1つ下のブロックで2、5、8ラインの出力を生成し、さらに1つ下のブロックでは3、6ラインの出力を生成する。このような3パターン

\*慮する必要がある。

【0055】例えば、縮小率「1/2」の場合には、1画素1ライン目を起点として垂直方向に沿って1から順に番号を付していった場合に、そのうちの奇数番号が付された画素位置について演算結果が出力される。

【0056】つまり、上記したWK1、PWK2、WK2についてはいずれも垂直方向において1、3、5、7ライン目(奇数番目の垂直ライン)の演算結果を生成するために、下記に示す演算パターンを使用すれば良い。

【0057】

5ライン目	7ライン目
CP5	Dummy
CP12	CP14
CP20	CP22

※【0059】

1ライン目	5ライン目
WK1: CP1	CP5
PWK2: CP8	CP12
WK3: CP16	CP20

20 縮小率「1/3」の場合には上記したように画素位置の数値が3パターン存在するので、下記のように3つの態様となる。

【0060】

Dummy
CP15 (=CP0)
CP22

Dummy
Dummy
CP23 (=CP7)

て1、4、7ライン用演算パターンを、PWK2について2、5、8ライン用演算パターンを、WK1について3、6ライン用演算パターンをそれぞれ選択する必要がある。

【0061】上記のように、縮小率の分母が整数の場合には演算パターンが比較的分かり易いが、例えば、縮小率「1/2.5」のように小数部「0.5」を有する場合には、0.5ライン分のずれに対する処理を必要とする。

【0062】縮小率「1/2.5」の場合には、前記し

たように画素位置の数列について全部で5通りのパターンがあることになるが、これに応じた演算パターンを全て挙げるのは繁雑であり、また、それほど益もないの\*

#### 演算パターン

WK1: (CP2+CP3); CP5, Dummy, Dummy

WK2: (CP7+CP8), CP10, (CP12+CP13), CP15

(=CP0)

WK3: CP16, CP18, (CP20+CP21), CP23 (

=CP7)

尚、「(A+B)」はパターンA、Bから得られる演算結果に対して後述のサブフィルタをかけることで0.5ラインずれた箇所での出力信号を得ること意味する。つまり、垂直方向に1ブロック当たり0.5ラインずつのずれがあることに注意を要する。

【0064】しかして、本発明に係る画像縮小処理方法においては、原画像データについて複数に分割されたブロックを単位にしてデータを転送するブロック処理を基本とし、当該データに対して水平又は垂直方向のフィルタ処理を行う際には、対象ブロック及びその近隣に位置するブロックについてそれぞれのフィルタ処理を行うための分割フィルタを用いて演算を行う。それから、フィルタ処理後の画素データについて画像縮小率に応じて規定される画素位置でのデータを縮小データとして出力する。

【0065】複数の分割フィルタについては、図5に示したように、フィルタ処理に使用する多数の演算パターン(CP0~CP23)を予め用意しておき、その後の縮小処理時には、設定された画像縮小率に応じて演算に必要な演算パターンを選択して各フィルタ毎に分割処理を行う。即ち、現ブロックよりも時間的に後に位置する後続ブロック内の画素について演算に必要な演算結果を先行して算出しておき、その後に上記後続ブロック内の画素について演算を行う際には、当該ブロックよりも先行して算出されている演算結果を受けとって、これを対象ブロック内の画素に係る演算結果と統合することにより最終的なフィルタ処理結果を出力する。尚、上記の説明では、垂直LPF処理においてフィルタ分割法を採用しており、処理に必要なワークデータ(WK1、WK2、PWK2)を先行して演算・作成しながら、現ブロックではワークデータWK2と、現ブロックでのワークデータWK3とを加算することで縮小結果が算出される。

【0066】図6は本発明に係る画像縮小処理装置の基本的構成例を示したものであり、画像縮小処理装置1は下記の構成要素を具備する(括弧内の数字は各要素の符号を示す)。

【0067】・水平処理部(2)

・垂直同時化部(3)

・垂直LPF処理部(4)

・垂直縮小部(5)。

\*で、垂直幅8画素に対して1、3、5、6ラインの出力を得る場合についてのみを示すに止める。

【0063】

【0068】尚、上記したように、原画像データについては複数に分割されたブロックを単位にしてデータが転送されるが、水平処理部2及び垂直LPF処理部4は、当該データに対して水平又は垂直方向のフィルタ処理を行うフィルタ処理手段6を構成している。

【0069】原画像情報(ソースデータ)は、まず、水平処理部2に送られて、内部の水平ローパスフィルタにより処理される。例えば、1ブロックの水平幅を32画素とし、これに対して15タップのフィルタをかける場合には現画素の他に14画素分の連続した画像データを集める必要がある。また、上記したように、縮小率が「1/2.5」のときのように「0.5」の端数が生じる場合には、画素重心をその分だけずらすための処理を行う必要がある。尚、水平処理部2内には、図示するように、輝度(Y)信号用の水平処理部2aと色(C)信号用の水平処理部2bが並設されている。

【0070】図7は水平処理部の基本構成例を示したものであり、輝度信号処理と色信号処理についてローパスフィルタのタップ数等、固有の箇所を除いて両者の回路が同形であるので、図ではそれらに共通する構成を示す。

【0071】縮小前の信号(データ)は、データ保持部7、水平ローパスフィルタ処理部(以下、「メインフィルタ」という。)8を経た後、ディレイ調整部9及びサブフィルタ10にそれぞれ送られる。

【0072】データ保持部7は、所定画素数のデータを保持するためにメインフィルタ8の前段に配置されている。輝度信号処理では、14画素×8ライン分のデータを保持できるようになっていて、例えば、現在処理中の現ブロックに対して、その左隣に位置するブロック内で右端側14画素の各データと、現ブロック内の最左端の画素データとを合わせて15タップのローパスフィルタに入力して出力を得ることができる。また、色信号処理では、28画素×8ライン分のデータを保持できるようになっていて、隣接画素28個と対象画素1個のデータをあわせて29タップのローパスフィルタに入力して出力を得ることができる。

【0073】メインフィルタ8内のローパスフィルタについては、例えば、輝度信号処理の場合に15タップ、色信号処理の場合に29タップのローパスフィルタがデジタルフィルタとして形成される。これは、既知のよ

うに色成分信号の場合には水平方向に2つのコンポーネント信号Cr、CbがY信号の隣に交互に存在する、つまり、各色信号が1画素おきに存在するため、ある縮小率におけるY信号のフィルタが、例えば、15タップ、

(a, b, c, d, e, f, g, h, g, f, e, d, c, b, a)であったときに、同一縮小率、同一フィルタ特性における色信号用ローパスフィルタについては、各要素の間に14個分の要素が挿入されて、(a, 0, b, 0, c, 0, d, 0, e, 0, f, 0, g, 0, h, 0, g, 0, f, 0, e, 0, d, 0, c, 0, b, 0, a)の、29タップ構成となるからである。

【0074】尚、メインフィルタ8には縮小率選択(あるいは指定)信号「IRATIO」(画像の縮小率を選択するための指定信号)が供給されるようになっており、当該信号及び後述のフィルタ切替タイミング信号に応じて、輝度信号処理用ローパスフィルタや色信号処理用ローパスフィルタの係数がそれぞれ異なる値となるように可変制御される。

【0075】サブフィルタ10は、縮小率が「1/2.5」や「1/3.5」等の場合に、0.5ライン分の画素ずらしのために設けられたフィルタであり、これには輝度信号について(1, 1)のフィルタを、色信号について(1, 0, 1)のフィルタを用いる。尚、(1, 1)フィルタは隣接画素(の輝度信号)同士にかかるフィルタであり、(1, 0, 1)フィルタは1つ飛びの画素(の色信号)同士にかかるフィルタである。この違いは、上記したように色信号において信号CrとCbが交互に1画素おきに出力されることに依る。また、縮小率が「1/2」や「1/4」等のようにきりの良い場合(選択すべき画素番号が整数値の場合)には、このようなサブフィルタ10を経ることなくディレイ調整部9で時間調整だけを行えば済む。サブフィルタの次数についてはより高次のものや、出力画質の向上を目的として重み付けの仕方を変更する構成等も考えられるが、構成や処理の簡単さを考慮すると上記の形態(サブフィルタによる演算を単純平均演算とする形態)が好ましい。

【0076】ディレイ調整部9及びサブフィルタ10の後段には切換部11が配置されており、該切換部はフィルタ切替タイミング信号「IFLSL」によって制御される。このタイミング信号「IFLSL」は装置内の図示しない信号生成部で作られ、輝度信号処理においては、当該信号がH(ハイ)レベルのとき(出力する画素重心が0.5ラインずれるとき)にサブフィルタ10の出力が選択され、当該信号がL(ロー)レベルのとき(出力する画素重心にずれがないとき)にディレイ調整部9の出力が選択される。また、色信号処理においては、信号「IFLSL」が、H(ハイ)レベルのとき(出力すべき画素の色信号の種類(Cr又はCb)が現画素の色信号の種類と違うとき)にサブフィルタ10の出

力が選択され、当該信号がL(ロー)レベルのとき(出力すべき画素の色信号の種類(Cr又はCb)が現画素の色信号の種類と同じとき)にディレイ調整部9の出力が選択される。

【0077】切換部11の後段には抜き出し部12が設けられており、これは前記した画素選択を行ったり、誤差の丸め処理等を行うものである。つまり、信号IRATIOで指定された縮小率に応じた画素位置の間隔をもって各画素のデータが抜き出され、例えば、縮小率「1/2」では画素番号が奇数番号の画素位置を選択してデータが出力される(この出力信号が水平縮小後の信号である。)

【0078】但し、色信号については出力後にCbとCrとが交互に配列されるように抜き出す必要がある。

【0079】図8は縮小率「1/2」における処理例を概念的に示したものであり、「IYC」が原画像入力(色信号)を、「OHL PFC」が色信号処理用ローパスフィルタの出力信号を、「OH SCLC」が抜き出し後の出力信号をそれぞれ示している。本例では、1画素分のCbとCrを飛ばして、CbとCrとを1組みにしてそのまま抜き出せば良い。

【0080】図9は縮小率「1/2.5」における処理例を概念的に示したものであり、「IYC」、「OHL PFC」、「OH SCLC」の意味は既述の通りである。本例では、CbとCrとを組みにしてそのまま抜き出す操作と、(1, 0, 1)フィルタをかけた後のCbとCrとを組みにして出力する操作とが、交互に繰り返される(尚、縮小率「1/3.5」等においても抜き出しのタイミングが異なるだけで、基本的には同じである。)

【0081】垂直同時化部3は、図6に示すように、輝度信号処理用バッファ3aと、色信号処理用バッファ3bを備えており、水平処理後の画素データに対する同時化手段を構成する。つまり、本回路部は、1ブロックに係る水平幅を画素単位で「a」個とし、1ブロックに係る垂直幅を画素単位で「b」個とするとき、「a × (b - 1)」個分の画素データを保持するものであり(最後の1ラインについては、入力時にリアルタイム処理を行えば良い。)、垂直方向におけるフィルタ処理手段(つまり、垂直LPF処理部4)の前に配置される。例えば、垂直同時化部3には、上記の処理を経て水平方向にローパスフィルタがかかった画像データが入力され、ここで垂直方向に8ライン分の同時化を行うが、これは、後述する垂直LPF部での演算パターンにデータを通す前の予備的処理として必要である。

【0082】図10は垂直同時化部3の構成例を示したものであり、2つの垂直同時化読出制御部13Y、13Cから構成されている。つまり、垂直同時化読出制御部13Yが輝度信号処理用とされ、これには上記した輝度信号用水平処理部2aの出力が入力され、他方の垂直同

時化読出制御部13Cが色信号処理用とされ、上記した色信号用水平処理部2bの出力が入力される。各読出制御部については16画素×8ラインの処理が行われ、8ライン分の(垂直)同時化されたデータがそれぞれ出力される。尚、各読出制御部には各種のタイミング制御信号が供給されて、データ読込みや書き込み、データ更新等のタイミング、1ブロック内の水平1ライン(32画素分)の認識タイミング等について制御される(例えば、各読出制御部に対してそれぞれ供給されるタイミング信号は、Y信号やC信号の有効タイミング入力を規定し、また、垂直同時化読出制御部13Yから出力されるタイミング信号は、同時化されたY信号及びC信号の有効タイミング出力を規定する。)

【0083】垂直LPF処理部4は複数の分割フィルタを有しており、それぞれのフィルタ処理に使用する多数の演算パターンが用意されており、縮小処理時には設定された画像縮小率に応じて必要な演算パターンが選択されて各フィルタ毎に分割処理される。本回路部には、例えば、図5に示した演算パターン(CP0~23)に対応する回路構成を含んでおり、図6に示すように輝度信号用垂直処理部4aと、色信号処理用垂直処理部4bを備えている。

【0084】図11は垂直LPF処理部内の垂直演算部14について構成例を示したものであり、輝度信号処理のために用意された3つの分割フィルタ15Y1、15Y2、15Y3からなる回路部15と、色信号処理のために用意された3つの分割フィルタ16C1、16C2、16C3からなる回路部16とを備えている。

【0085】即ち、回路部15については、各分割フィルタに対して垂直同時化読出制御部13Yの出力データがそれぞれ供給されるが、上段の分割フィルタ15Y1は、2ブロック先での演算に必要な演算結果(WK1)を得るためのフィルタ(前記CP0~CP6を参照。)、中段の分割フィルタ15Y2は、1ブロック先での演算に必要な演算結果(PWK2)を得るためのフィルタ(前記CP7~CP15を参照。)、下段の分割フィルタ15Y3は、現ブロックでの演算に必要な演算結果(WK3)を得るためのフィルタ(前記CP16~CP23を参照。)である。

【0086】また、回路部16については、各分割フィルタに対して垂直同時化読出制御部13Cの出力データがそれぞれ供給されるが、上段の分割フィルタ16C1は、2ブロック先での演算に必要な演算結果(WK1)を得るためのフィルタ(前記CP0~CP6を参照。)、中段の分割フィルタ16C2は、1ブロック先での演算に必要な演算結果(PWK2)を得るためのフィルタ(前記CP7~CP15を参照。)、下段の分割フィルタ16C3は、現ブロックでの演算に必要な演算結果(WK3)を得るためのフィルタ(前記CP16~CP23を参照。)である。

【0087】尚、図11において各分割フィルタには前記信号IRATIO、IFLSL、タイミング信号(図10の垂直方向読出制御部13Yの出力するタイミング信号)がそれぞれ入力され、また、分割フィルタ15Y1~3はタイミング信号をそれぞれ出力するが、これらは垂直LPF後の有効タイミング出力を規定する。

【0088】今仮に設定された縮小率が「1/2」であるとする、フィルタ後の出力については、ブロックの垂直方向で1、3、5、7画素目に当たるデータを抽出して並べれば良い。そこで、下記に示す(i)乃至(ii)の手順を踏む(図12の右側に示す3種の分割フィルタを参照。尚、各演算パターンについては図5を参照。)

【0089】(i) 2ブロック下に位置するブロックのためのワークデータWK1用には演算パターンCP1を選択し、その演算結果を(外部)バッファ部へ出力する  
(ii) 1ブロック下に位置するブロックのためのワークデータPWK2用には演算パターンCP8を選択し、その演算結果を(外部)バッファ部へ出力する

(iii) 現ブロックのワークデータWK3用には演算パターンCP16を選択し、その演算結果を後段の垂直縮小部5に送出する。

【0090】尚、各分割フィルタには、上記した縮小率選択信号「IRATIO」が送られて上記したように縮小率に適合した演算パターンが使用され、また、フィルタ切替タイミング信号「IFLSL」が送られて、上記した0.5ラインの画素位置ずれに係る処理が行われる。

【0091】垂直縮小部5は、図6に示すように、ワークデータ生成部5aと、輝度信号用垂直縮小部5b、色信号用垂直縮小部5cを備えており、後2者が演算処理手段を構成している。

【0092】前記したように、複数の分割フィルタについては、各フィルタ処理に使用される各種の演算パターンが用意されており、それぞれの演算パターンによって算出されるワークデータに基づいて縮小後の出力が計算される。つまり、各分割フィルタによる演算処理時には、現ブロックよりも時間的に後に位置する後続ブロック内の画素について演算に必要な演算結果を先行して算出しておくことが求められ、そのためにワークデータ生成部5a(ワークデータ生成手段)が必要となる。

【0093】そして、ワークデータ生成部5aによって現ブロックよりも先行して算出されたデータは、輝度信号用垂直縮小部5bや色信号用垂直縮小部5cに入力され、それらの回路部は、ワークデータ(演算結果)を受けとって、これを現ブロック内の画素に係る演算演算と統合することにより最終的なフィルタ処理結果、即ち、縮小結果を出力する。

【0094】例えば、各垂直縮小部は、3分割フィルタにより得られるワークデータ(演算結果)の処理及び縮

小データの生成を行い、最終的に縮小率に応じた画素選択の結果を出力する。

【0095】図12の左図は、ワークデータ処理及び縮小データの生成処理の流れを概念的に示したものである。

【0096】図中のWK1演算部17は、上記分割フィルタ15Y1や16C1を含む部分であり、同様に、PWK2演算部18は、上記分割フィルタ15Y2や16C2を含む部分、WK3演算部19は、上記分割フィルタ15Y3や16C3を含む部分である。

【0097】前記したように、中間処理結果を示すワークデータWK2は、上式「 $WK2 = PWK2 + WK1$ 」から分かるように、WK1を必要とする。つまり、WK2を得るためには、PWK2の演算後にWK1を必要とするので、そのためにWK1演算部17により得られるWK1を（外部）バッファ部20に一時的に保持しておく必要がある。

【0098】同様に、最終演算結果SCLは、上式「 $SCL = WK3 + WK2$ 」から分かるように、WK2を必要とする。つまり、SCLの算出には、WK3の演算後にWK2を必要とするので、そのためにWK2を（外部）バッファ部20に一時的に保持しておく必要がある。

【0099】図12において、「OWK」はバッファ部20に対する出力ノードを、「IWK」はバッファ部20からの入力ノードを、「OYC」は縮小データの出力ノードをそれぞれ示しており、WK1が先ず、「OWK」からバッファ部20に一旦格納された後、「IWK」から取り出されて、加算部21に送られ、ここでPWK2と足し算されてWK2が得られる。そして、このWK2が「OWK」からバッファ部20に一旦格納された後、「IWK」から取り出されて、加算部22に送られ、ここでWK3と足し算されてSCLが得られ、これが「OYC」から出力されるという流れになる。

【0100】上記した垂直縮小部5b、5cは、このように、先行計算結果と現在の計算に基づいて演算処理を行う演算処理手段を具備している。

【0101】図13は垂直縮小部の構成例23を示すものであり、輝度信号処理と色信号処理について同形の回路構成を採るので、以下では両信号をひとまとめにして説明する。

【0102】先ず、ノード「IWK」には、図示しないバッファ部から垂直処理用のワークデータ（WK1、WK2）が入力された後で、ビット調整部24に送られ所定ビット（例えば、Y信号を8ビットから12ビット化し、C信号を8ビットから11ビット化する。）にしてデータ（これを「WK1」と記す。）を出力する。

【0103】ノード「IVSCL1」には、現時点での演算結果であるワークデータWK1が供給され、これは出力順序調整部25にそのまま送られて、所定のビット

数（例えば、16ビット）のデータに変換された後に、ノード「OWK」を介して、図示しないバッファ部に送られて一時的に格納される。

【0104】上記したデータWK1は、分離部26で振り分けられ、WK1については加算部27に送られ、またWK2については加算部28に送られる。

【0105】加算部27に入力されるWK1は、現時点でのワークデータWK1からみれば、1つ上のブロックに係る演算結果であり、ノード「IVSCL2」からPWK2が入力された時点でPWK2と加算されて、その結果はWK2となり、出力順序調整部25に送られる。つまり、出力順序調整部25はノード「IVSCL1」からWK1を受けたときにはそのままWK1をノード「OWK」に出力し、加算部27からWK2を受けたときにはこれをノード「OWK」に出力するので、バッファリング後の「WK1」は1つ前の時点でのWK1又はWK2である。

【0106】ノード「IVSCL3」からのWK3は加算部28に送られ、ここでは上記分離部26からのWK2と足し算される。つまり、このWK2はWK3からみて1つ上のブロックに係る演算結果であり、両者を加算した結果が上記SCLであって、これがノード「OYC」から出力される。

【0107】尚、加算部27、28は、加算演算の他、丸め処理やビット上限を規制する処理等も行う。

【0108】また、実際にはこのような回路が輝度信号用と色信号用の回路としてそれぞれ並設されるため（図6参照。）、Y信号、C信号を合わせてノード「OWK」に出力させるための処理（ダミービットの付加等）を要する。

【0109】最後に、画端処理について説明する。

【0110】画像縮小を行う場合に、画面周辺に位置する画素について如何に処置するかが問題となる。即ち、このような場所では、フィルタ処理に必要な数の近隣画素をそのままでは確保できないからである。そこで、画像周囲でのレベルを一律にゼロとみなして処理する方法が考えられるが、これでは画像周辺部において処理が破綻して画質を著しく劣化させる虞がある。

【0111】これを回避するには、画像周辺における原画像の構成画素に対するミラー画素（即ち、画像周辺の境界部に鏡を置いたと仮想した場合に、鏡に映った原画像の画素を、画像周囲の画素として配置させる方法で得られるもの。）等、ダミー画素を画像周囲に配置させる方法が挙げられるが、処理の簡単化という観点からは、画像の周縁（画端部）の画素データを使って画素領域を外方に拡大する方法を採用することが好ましい。

【0112】図14は画端処理の説明図である。

【0113】同図において四角枠RAで囲んだ領域が原画像の領域を示し、その外側の四角枠RBと四角枠RAとの間に示す領域が外方に拡張された領域であり、本発

明では、当該拡張領域におけるレベルをゼロとするのではなく、丸枠で囲んだ部分を拡大して示すように、画面端の画素データを保持したまま外方に領域を拡張する。

【0114】つまり、図には入力される原画像領域の画端に位置するそれぞれの画素（ダミー画素）について、そのレベルを数値例として示しており、例えば、左端の垂直方向については、上から「12」、「5」、「4」、「3」、「1」のレベルとされ（数値そのものには特別な意味はない。）、これらが所定の画素数分に亘って左方に複写された形で保持されている。同様に、下端の水平方向については、左から「1」、「4」、「3」、「2」、「5」のレベルとされ、これらが所定の画素数分に亘って下方に複写された形で保持されている。尚、画面隅の画素（レベル「1」）については、その左方及び下方に延長された部分をそれぞれ2辺とする四角形（正方形）領域内においてレベルが全て「1」とされている。

【0115】水平処理部2において、所定のタイミング信号により処理の対象ブロックが画像端を含むブロックであることが認識された場合には、水平方向における所定画素数（例えば、8画素）のダミー画素を演算パターンに対して一度に入力することにより、当該ダミー画素を含む水平フィルタ処理を簡単に行うことができる。

【0116】また、垂直LPF処理部4においても、ワークデータの算出に必要な画像端でのダミー画像についてそれらのレベルが一樣であること、つまり、上記したように同じレベルの画素が垂直方向に沿って配列されるため、これらを演算パターンにかけることで容易にフィルタ処理を行える。つまり、画像端処理に際して周辺画素（最も端の画素）のレベルをホールドし、当該レベルと同じレベルのダミー画素を用いてこれが連続して並んでいるかのように見せかけて、周辺領域を外方に拡張することによってフィルタ処理が非常に簡単なものとなる。

【0117】尚、画像端でホールドする画素数（ダミー画素数）について、水平処理では、対象ブロックの最左端（又は最右端）の画素位置がタップ数の中央位置となるように調整する。例えば、輝度信号に15タップフィルタを用いる場合には、その中央位置が「8」であるので、最左端（又は最右端）の画素位置より左側（又は右側）に7画素分のダミー画素をホールドして処理すれば良い。同様にして、色信号に29タップフィルタを用いる場合には、その中央位置が「15」であるので、最左端（又は最右端）の画素位置より左側（又は右側）に14画素分のダミー画素をホールドして処理すれば良い。そして、垂直処理では、前記の例においてそれぞれ8画素分の3つの分割フィルタの演算パターンに必要なのが最大3ブロックに亘る画素データであることから分かるように、垂直方向におけるダミー画素数は1ブロックの垂直幅であり、輝度信号、色信号ともに8画素である。

つまり、対象ブロック内の画素位置がタップ数の中央位置となるように調整し、例えば、15タップフィルタの場合には、最上端位置のブロック（図1の第0ブロックライン）内の最上ラインの画素位置が8番目のタップ位置に来るようにするためには、当該ブロックラインの上に7画素分以上のダミー画素のブロックからなるダミーブロックラインを確保すれば良い（実際には0.5ライン目の処理を考慮して8画素分のダミーデータを確保する。）。また、最下端位置のブロック（図1の第「Y/b-1」ブロックライン）内の最下ラインの画素位置が8番目のタップ位置に来るようにするためには、当該ブロックラインの下に7画素分以上のダミー画素のブロックからなるダミーブロックラインを確保すれば良い（実際には0.5ライン目の処理を考慮して8画素分のダミーデータを確保する。）。

【0118】図15は、画端処理の流れを概念的に示したものであり、処理対象となるブロックが画像端の画素位置を含む場合には、当該画素位置の周囲にダミー画素を付設することにより画像周辺域を拡張するために周辺拡張手段29を設けている。

【0119】即ち、原画像データは、周辺拡張手段29を経ることによって周辺域に上記したダミー画像が付加されて処理対象の画素域が外方に上げられる。その際には、上記したように、画像端の画素位置の画素レベルと同じレベルのダミー画素を付設するのが最も簡単である。原画像データ及びダミー画像を含む全データは、その後、フィルタ処理手段6に送られてローパスフィルタによる演算が施される。例えば、水平処理では上記した対象画素を中心とする所定タップ数のフィルタ処理が行われ、また、垂直処理ではダミー画素を含む隣接ブロック及び対象ブロックに対してそれぞれに対応する分割フィルタにより演算処理が行われる。尚、周辺拡張手段29についてはメモリ等の記憶手段を用いたデータの書き込み、複写、読み出し等の処理によって容易に実現することができるので、回路構成が著しく複雑化するという弊害がない。

【0120】上記した装置及び方法によれば、下記に示す種々の利点が得られる。

【0.1.2.1】・ブロック処理を採用することにより、画像縮小処理時に必要なメモリ容量を減少させ、低コストで画像処理システムを構築することができる・特にJPG圧縮や伸長のようにブロック単位で画像処理を行う場合に上記方法は相性が良く、システム全体の処理速度を向上させるのに適している・従来の画像端処理では、処理結果の画質を損なう可能性があったが、上記したように画像周辺を画像端位置の画素と同レベルのダミー画素を付加して領域拡張を図ることで画質劣化を防止するとともに、フィルタ処理が簡単で済むようになる。

【0122】

【発明の効果】以上に記載したところから明らかなよう

に、請求項 1 や請求項 14 に係る発明によれば、ブロック処理の採用により容量の大きなメモリを必要とせず、また、現ブロック及びその近隣ブロックに対するフィルタの分割処理によって画像縮小処理の処理速度について高速化を実現することができる。

【0123】請求項 2 や請求項 15 に係る発明によれば、複数の区分けされた分割フィルタについて、多数の演算パターンを予め用意しておき、縮小処理時には、画像縮小率に応じて演算に必要な演算パターンを選択して各フィルタ毎に分割処理を行うことで演算を効率的に実行できる。

【0124】請求項 3、4、16 に係る発明によれば、分割フィルタによる分割処理時に、先行して算出される演算結果を受け取って、これを対象ブロック内の画素に係る演算結果と統合することにより、処理の高速化及び効率化を図ることができる。

【0125】請求項 5 乃至 8、17 乃至 19 に係る発明によれば、処理対象となるブロックが画像端の画素位置を含む場合に、当該画素位置の周囲にダミー画素を付設して処理することにより、画像端での破綻を防止し、画質劣化を防止することができる。

【0126】請求項 9 乃至 12、20 乃至 22 に係る発明によれば、画像端での画素位置の画素レベルと同レベルのダミー画素を付設することにより、簡単に周辺域を外方に拡張することができる。

【0127】請求項 13 に係る発明によれば、同時化手段を、垂直方向におけるフィルタ処理手段の前に配置することにより、演算パターンに渡す画素データの組を同時化し、処理の正確さを期すことができる。

【図面の簡単な説明】

【図 1】本発明に係るブロック処理についての説明図である。

【図 2】図 3 とともに縮小処理における選択画素についての説明するための図であり、本図は縮小率「1/2」の場合を示す。

【図 3】縮小率「1/2.5」の場合を示す説明図である。

【図 4】分割例を示すパターンの説明図である。

【図 5】最大 3 分割の 15 タップフィルタについて現ブロック内の 8 画素が採り得る位置の組み合わせの数をすべて網羅した演算パターンを示した図である。

【図 6】本発明に係る画像縮小処理装置の基本的構成例を示す図である。

【図 7】水平処理部の構成例を示す図である。

【図 8】色信号について縮小率「1/2」における処理例を概念的に示した説明図である。

【図 9】色信号について縮小率「1/2.5」における処理例を概念的に示した説明図である。

【図 10】垂直同時化部の構成例を示す図である。

【図 11】垂直演算部について構成例を示す図である。

【図 12】ワークデータ処理及び縮小データの生成処理についての説明図である。

【図 13】垂直縮小部の構成例を示す図である。

【図 14】画端処理の説明図である。

【図 15】画端処理の流れを概念的に示す図である。

【図 16】従来の処理例についての説明図である。

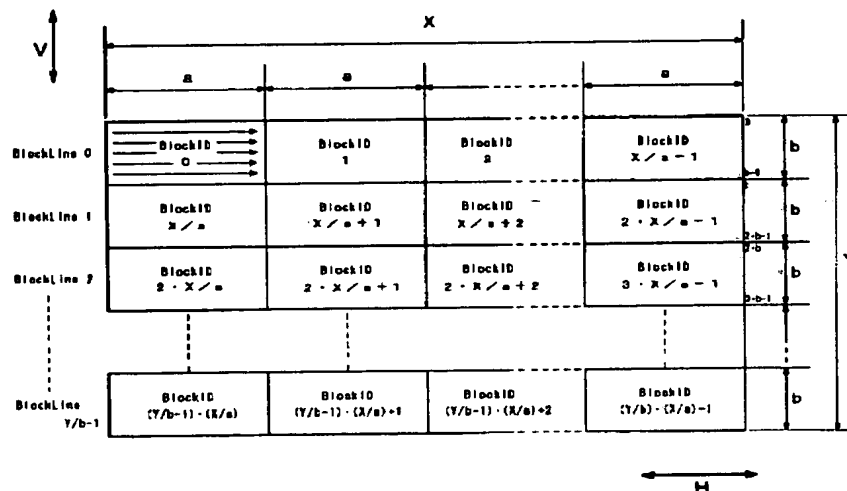
【符号の説明】

1…画像縮小処理装置、5a…ワークデータ生成手段、

6…フィルタ処理手段、15、16…分割フィルタ、1

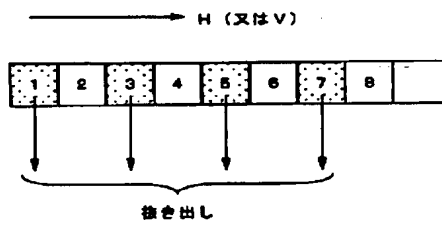
9…周辺拡張手段

【図 1】

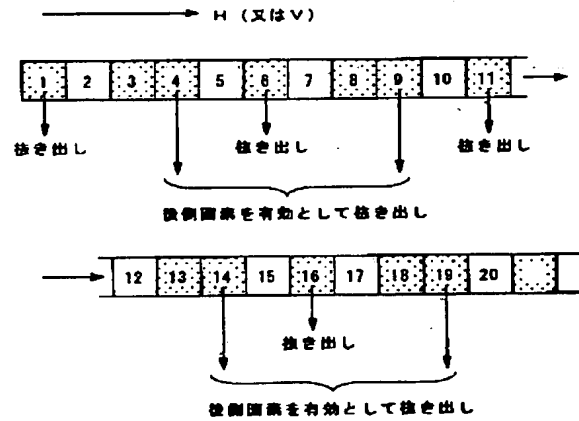




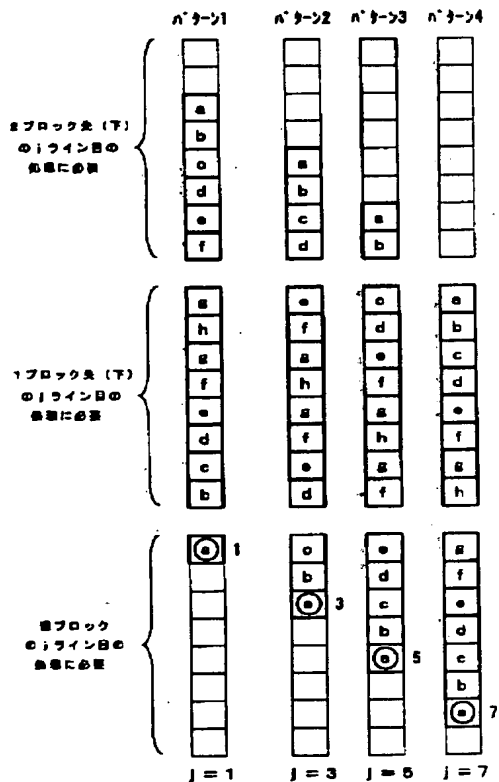
【図2】



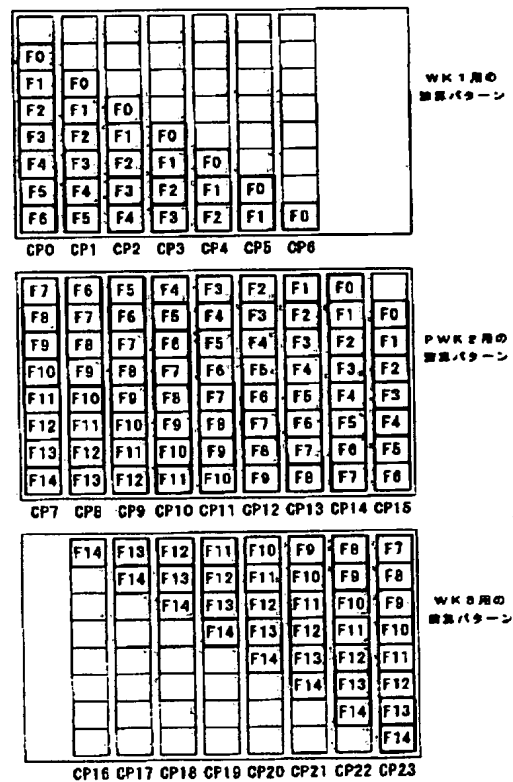
【図3】



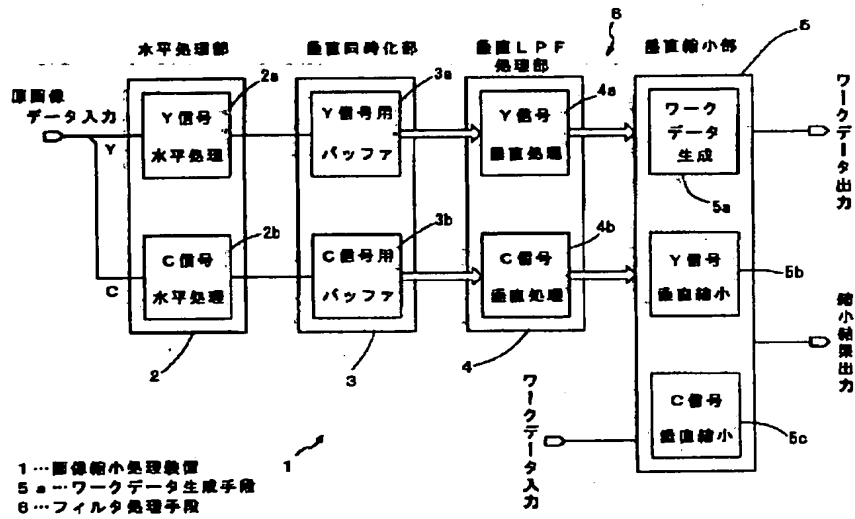
【図4】



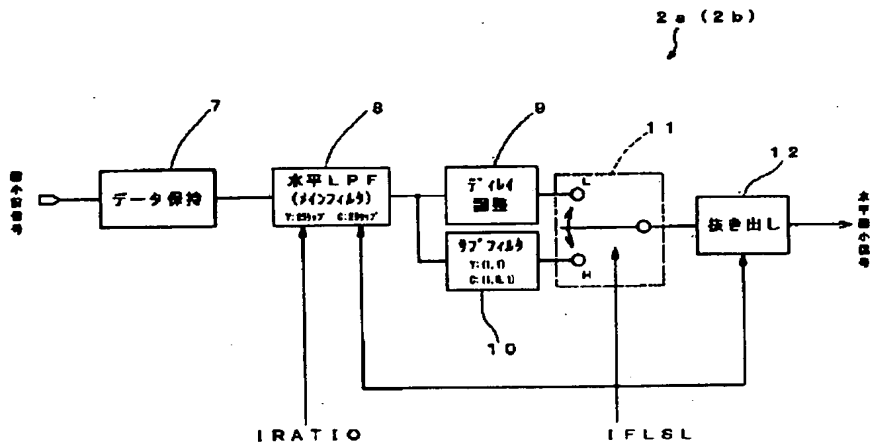
【図5】



【図6】

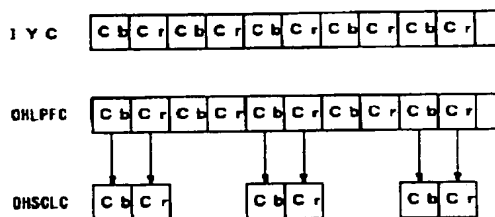


【図7】



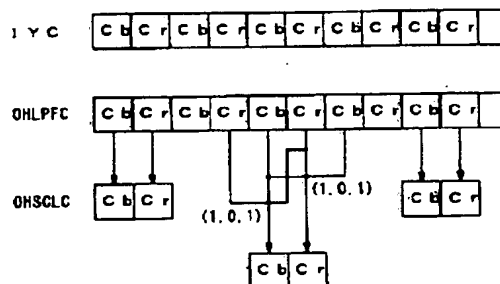
【図8】

縮小率 = 1/2

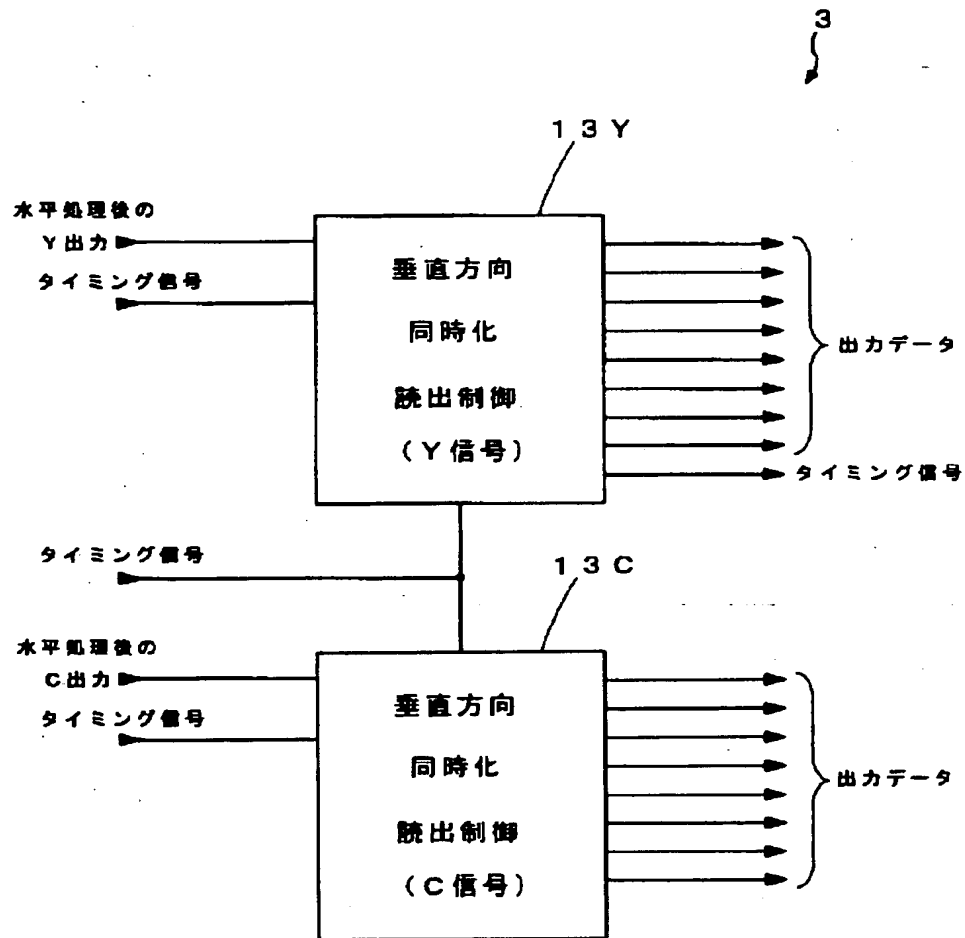


【図9】

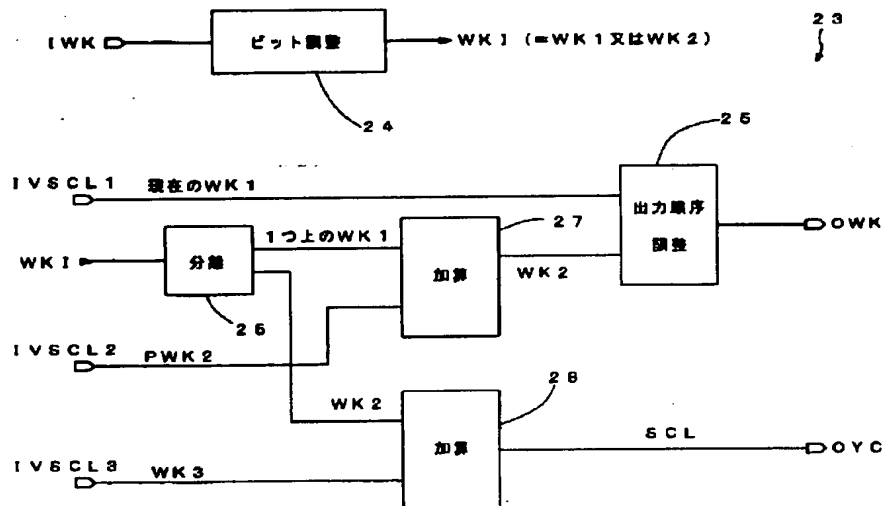
縮小率 = 1/2.5



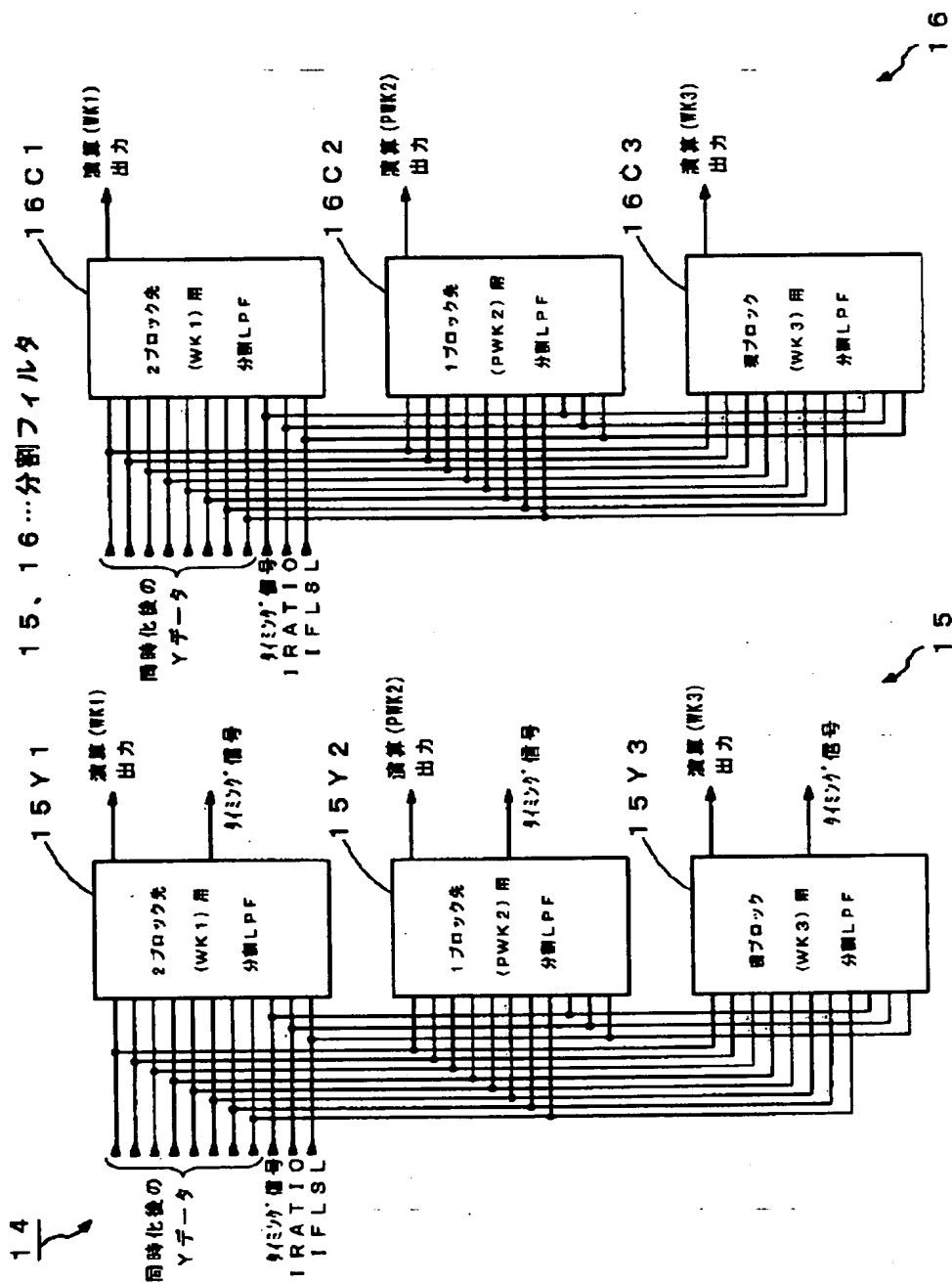
【図10】



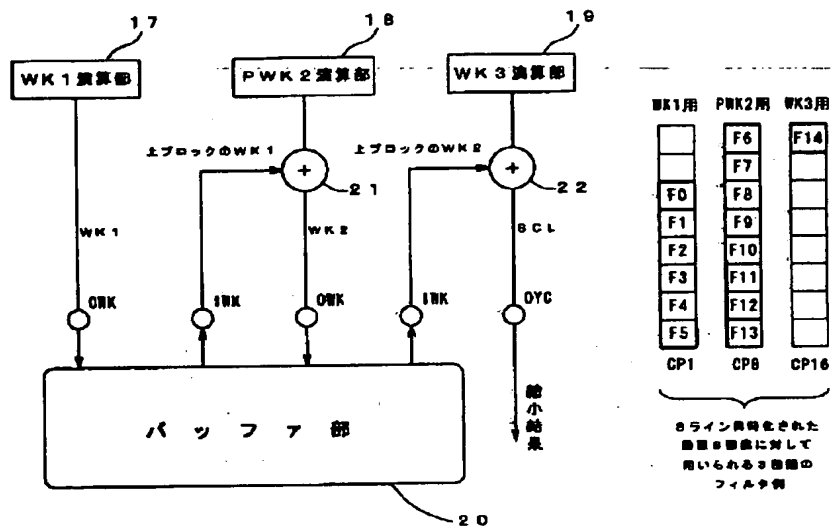
【図13】



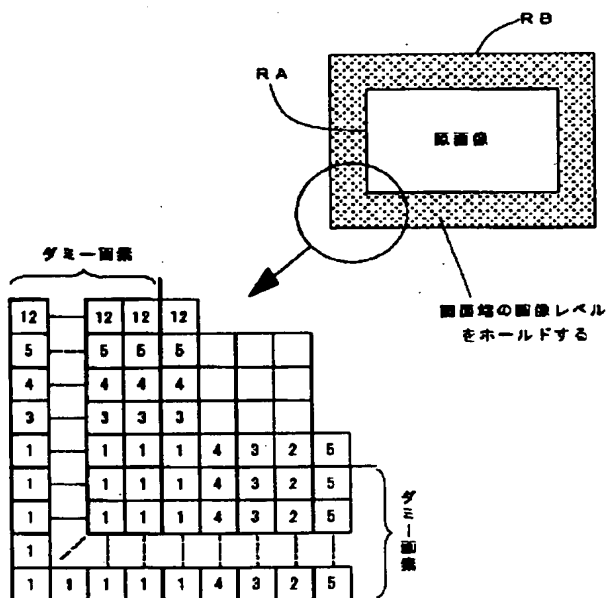
【図11】



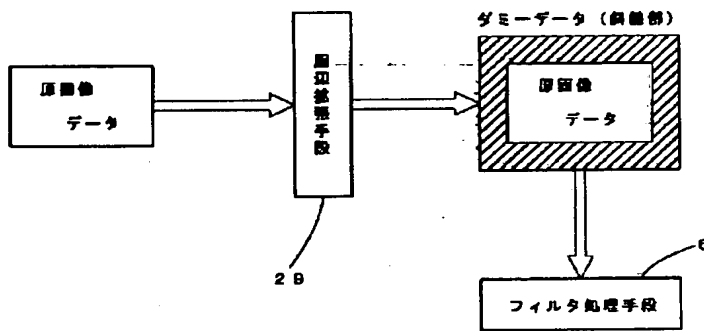
【図12】



【図14】

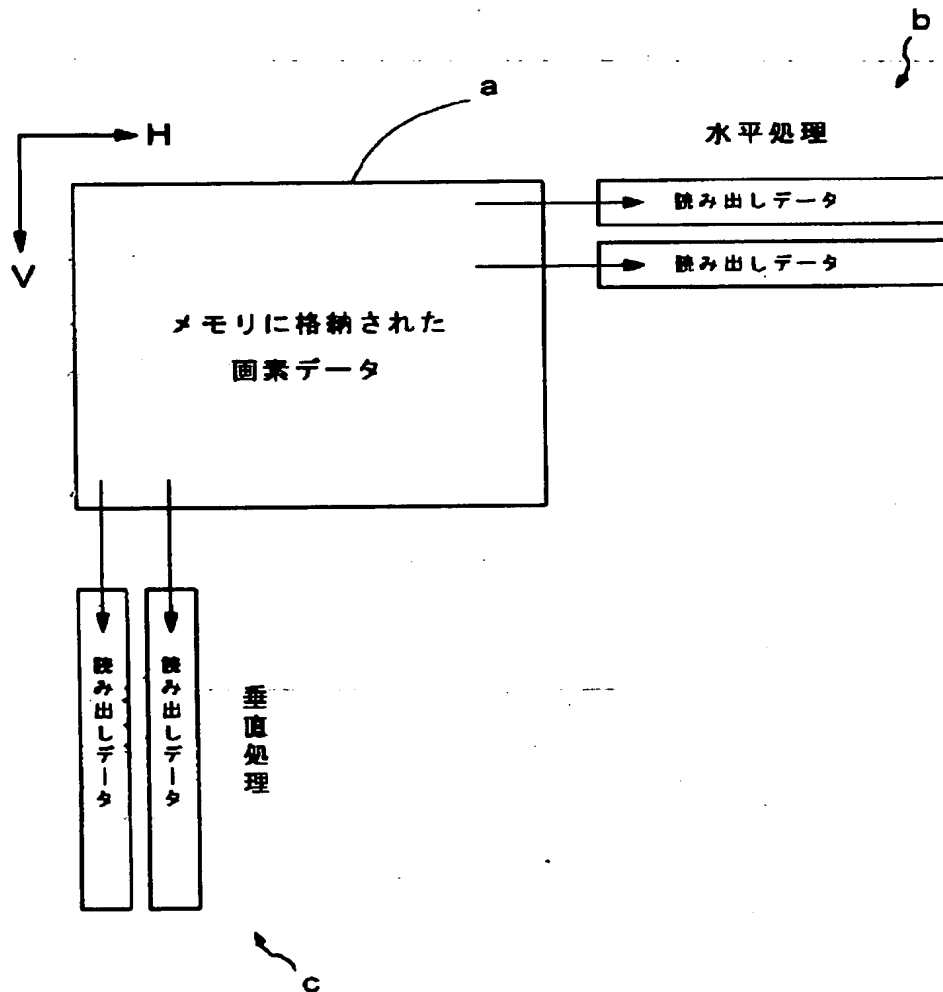


【図15】



8…フィルタ処理手段  
19…周辺処理手段

【図16】



フロントページの続き

Fターム(参考) 5B057 BA23 CD07 CE06 CH18  
5C076 AA22 AA36 BA05 BA06 BB06  
BB13 BB40 CB01  
5C077 LL17 LL18 MP08 PP01 PP04  
PP20 PP34 PP68 PQ08 PQ12  
PQ18 RR18